Dynamic Proxy Selection for Group Smartphone Energy Saving

Undergraduate Research Final Thesis

Presented in Partial Fulfillment of the Requirements for Graduation with Research Distinction in the Undergraduate School of The Ohio State University

By

Benedict Kok Wei Lee

Undergraduate Program in Electrical and Computer Engineering

The Ohio State University

2014

Dissertation Committee:

Dr. Xiaorui Wang, Advisor

Abstract

Smartphones and tablets are becoming more and more closely integrated with our daily lives. Smart devices have rapidly substituted traditional computers when it comes to performing certain daily activities such as web-browsing, email services, news feed, and etc. However despite the popularity of smart devices in recent years, smart devices continue to have a major critical flaw in their battery life. Several studies had been carried out to optimize energy consumption of individual smartphone through power management (managing GSM, Wi-Fi, and GPS), workload migration (cloud computing), and consolidation (virtual machine). However, unlike previous studies that focus on single phone energy saving methods, we propose a unique method whereby multiple smartphones collaborate to reduce energy consumption. In a phone grouping event, a proxy selection framework is vital to ensure that the workload can be delegated among all users, and that is delegated via a fair system. In this paper, a proxy selection framework is designed to ensure a self-sustained, fair, and efficient system that monitors user's characteristics and performs maintenance operations dynamically. This framework is implemented on real smartphones to experiment on various proxy selection techniques and to verify its overhead and energy saving. Real world traces is used to test the framework on different social activity scenarios. By using this framework, we can save up to 75% of energy in comparison to a normal usage of individual phones. However, there are two main limitations to this framework. The first is that the framework will not

be able to operate with the design of this system as a background app. Secondly, there is also an issue of privacy since usernames and passwords will be passed around between the smartphones. Nevertheless, even though the current study did not demonstrate a promising or reliable usage of this application, it provided some preliminary data for research direction as well as a good motivation and framework for future work.

Acknowledgments

I would like to express my deepest appreciation to Dr. Xiaorui Wang, who had convincingly conveyed a spirit of creativity and adventure in regard to research. His support and persistent help has been my main motivation in order for me to complete my research project. In addition, a huge thank you to everyone at the Power-Aware Computer System (PACS) Laboratory especially Macro Brocanelli for his technical assistance, constructive advice, and moral support.

I would also like to thank the College of Engineering at The Ohio State University for giving me the opportunity to pursue my research interest and also the Undergraduate Research Scholarship.

I wish to thank Dr. Yuan F. Zheng for being my oral exam committee and his great advices for the future work and improvement for this research.

Lastly, a special thanks to my parents Mr. and Ms Lee for their love, support, and encouragement in my study.

Vita

2003...Malaysia Certificate of Education,
Ken Hwa High School

2010...American Degree Transfer Program,
Sunway University College

2012 to present ....Department of Electrical and Computer Engineer, The Ohio State University

Fields of Study

Major Field: Electrical and Computer Engineering

Table of Contents

List of Figures

Chapter 1: Introduction and Motivation

.

I. Introduction

With the rapid development of smartphone technology, public consumers are expecting faster mobile processors, thinner devices, higher screen resolution, and new functionalities with each subsequent release. While processing power and performance of smart devices have been rapidly improving, innovation in battery technology has stagnated. The advancement in so many areas has caused the energy consumption to surpass the potential of battery capability. The effectiveness of energy consumption is one of the largest limiting factors of the future growth of smart devices.

A common trick to extend battery life is to disable wireless capabilities such as Wi-Fi, and GPS, when they are not in use. However, being able to constantly connect to the internet has become our daily life essential. Based on real-world trace study, most users waste phone power even though they are not interacting with the device, due to periodic network activities when checking social media notification or downloading new emails. Additionally, power is also wasted when users consistently read new but useless notifications.

Several research studies discuss the possibility of reducing power consumption through power management (managing GSM, Wi-Fi, and GPS), workload migration (cloud computing), and consolidation (virtual machine). Chon et al. [1] investigate user's daily patterns to predict user behavior, allowing better management of GSM, Wi-Fi, and

GPS. Workload migration is another common solution used to save energy. Chun et al. [2] investigate on using a combination of static analysis and dynamic profiling to offload partition application to the cloud, thereby optimizing execution time and saving energy. Several studies attempted to use virtual machines to reduce power consumption. Das et al. [3] use a technique known as desktop virtualization to migrate essential computer operations to a server, allowing the user to save energy by switching their desktop to idle. All these previous studies focused on a single phone energy saving methods. Unlike previous studies, this research uses a unique method whereby multiple phones work together to reduce energy consumption.

Since user's activities are common across different smartphones, a group of phones could potentially collaborate to reduce overall energy consumption. When a group of people geographically join a particular group event (e.g., meeting, dinner, party), it is most likely that the group may have a lot of similar application and features. The concept of consolidation allows a maximum amount of unused devices to switch to low power mode by migrating similar activity features to a proxy device. The main idea of this research is to select and rotate a user-device (the proxy) to perform the job of checking new notifications for the rest of the devices in the group (the clients), allowing the client to be placed in a lower power state to save battery in a fair and efficient manner.

This study is therefore intended to investigate the consolidation process within a group of smartphones to ensure the fairness of task delegation and high energy efficiency.

Objective

This research intends to achieve the following goals:

- Investigate various proxy selection techniques to maximize energy saving.
- Develop an algorithm that can rotate the role of the proxy device in a network while maintaining task delegation fairness and energy efficiency
- Design a proxy selection framework to ensure a self-sustained system that shifts activities processing and performs maintenance operation dynamically.

II. Motivation

To save energy consumption for a group of smartphones and to improve overall energy efficiency, we first need to determine whether multiple phones have similar activities that may lead to significant energy usage and can be consolidated. Based on a trace study of user activity [4], this trace provides us data about the app access time, duration, and app name of 34 users during one year. Figure 1 shows the popularity of top seven apps of 34 users in one year.

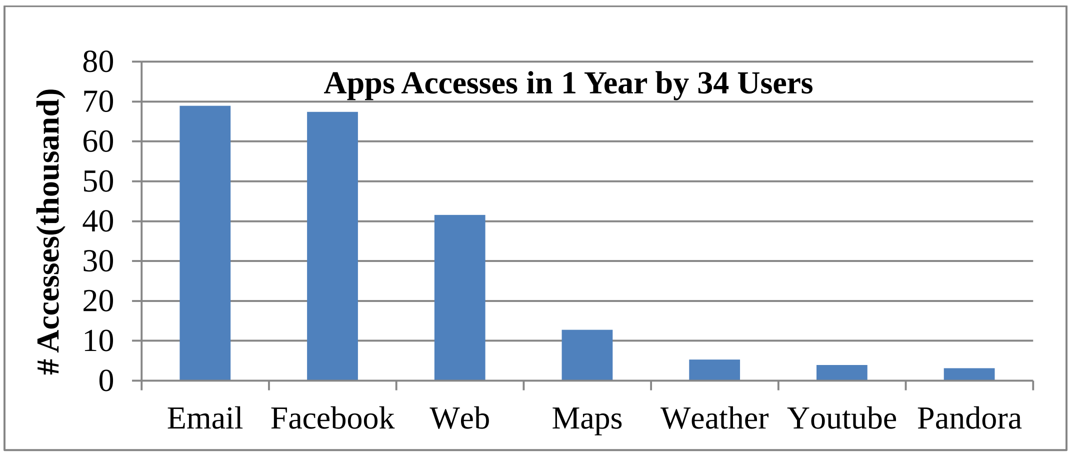

Figure 1. Top seven apps accessed by 34 users in one year trace study.

The access frequency of the top two apps, Email and Facebook, are almost twice as much as the third place app, web browsing and one magnitude larger than any other apps. In fact, by only focusing on the first two apps, they represent 67% of the user's interactions with the phone, which shows that there is a large possibility that different phones do have common activities. Furthermore, in order to save power consumption, we need to investigate the possibility for phones to offload most of their activities and turn themselves to an even lower power state than default standby power state. Figure 2 shows the duration of the app when it is active for both Email and Facebook app to study the user behavior.

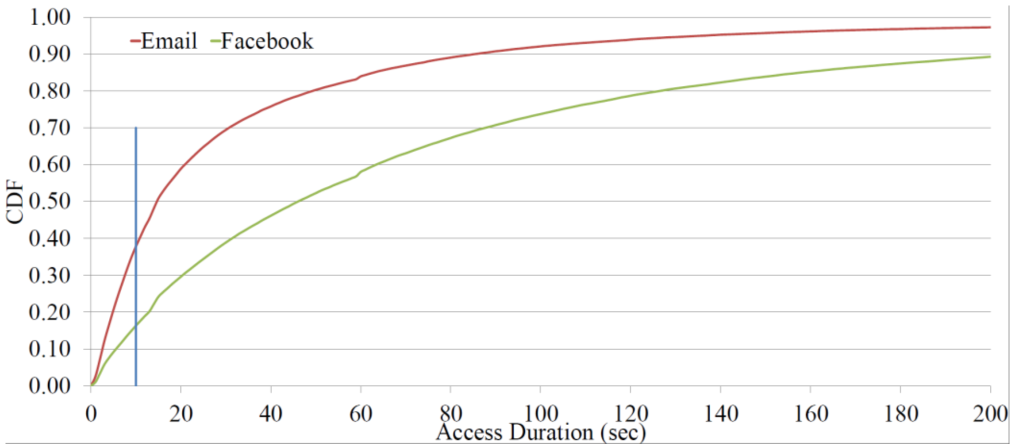

Figure 2. CDF of app active duration for Email and Facebook

For Email, about 40% of the activity duration are less than 10 seconds. We set this as the threshold because it is reported by Wasserman [5] that workers delete 106 spam emails on a average day and each deletion takes around 10.6 seconds to complete even with spam filter. Not to mention that some emails are unimportant or meaningless that the phone could waste more energy if users spend their time checking those emails. Thus, if there exists a way to allow users to know that the received email is an unimportant email before waking the entire phone to receive it, it could leave the phone to sleep longer and further save energy.

On the basis of the detailed analysis, we find that (1) Email and Facebook apps dominate over 67% of the total phone usage which is shown in Figure 1 and (2) about 40% of user interactions with those apps lead to wastage of power consumption which is shown in Figure 2. These findings suggest that it is sufficient to only concentrate on consolidating the most popular apps which shows promising results. In summary, we find that multiple phones could have common group features and significant energy saving can be achieved by reducing meaningless operations and let most phones enter into a lower power state.

Chapter 2: Methodology

I. Design Protocol

The main idea of this framework is to conserve energy by consolidating the activities of a number different of smartphone into one phone. Before we proceed to the technical details of the framework protocol here are some important definitions:

- Group event: Collaboration among a group of smartphones that are geographically near to each other to establish a network connection.
- Proxy-Client Mode: The users enter the proposed model service to prolong their battery life, when a group event is present.
- Proxy: A selected phone which receives activities to manage from clients.
- Client: All users that join the group operated under snooze state except for the proxy.
- Snooze State: A power state that operates with lowest possible CPU power state and basic GSM components.

In general, the whole system can be divided into two modes: proxy mode and client mode. In the system, there will only be one smartphone serving as a proxy from time to time while the other smartphones (Clients) operate in a snooze state. The proxy will be responsible for handling communication with the remote servers for each specific app. The whole architecture of the framework is shown below (Figure 3).

Figure 3. The framework architecture of proxy-client mode

There are basically four essential communications protocols in this framework as shown in figure 1 where each step of the protocol is marked with a number label.

The first step is to establish a network among the group of smartphone by sending a request message from the proxy phone for service consolidation. The initial network among phones will be established via Bluetooth. The reason Bluetooth is chosen is because it has very low energy consumption overhead in comparison with 4G and WiFi [6]. Even though, Bluetooth communication has a range of about 30 meters only it is still large enough to cover a group location such as a party place or conference room. At this stage, after the Bluetooth pairing of devices is complete the proxy will request for the user (client) phone number, app name, user account, and account password. The proxy will have a log of all its client details stored in the memory which is indexed in a hash table for quick lookup. After all the required data is provided to the proxy, the client will be put under snooze state. In this snooze state, the client can still receive message from the proxy.

In the second stage, the proxy will handle all the message exchange, services, and notification checking on behalf of the clients using remote services such as WiFi and 4G. The proxy is designed to periodically wake-up to check for new notification based on each account from its clients. The third stage takes place whenever there is new information being pulled from certain apps, whereby the proxy will send a notify message via short message service (SMS) to the target client. By sending the SMS it will wake up the client to inform the user about the new notification, the user can then decide whether to wake up the entire phone to interact with the app or stay in a snooze state. The last step is for the client to rejoin the network by sending a confirmation message to the proxy to resume its status at snooze state.

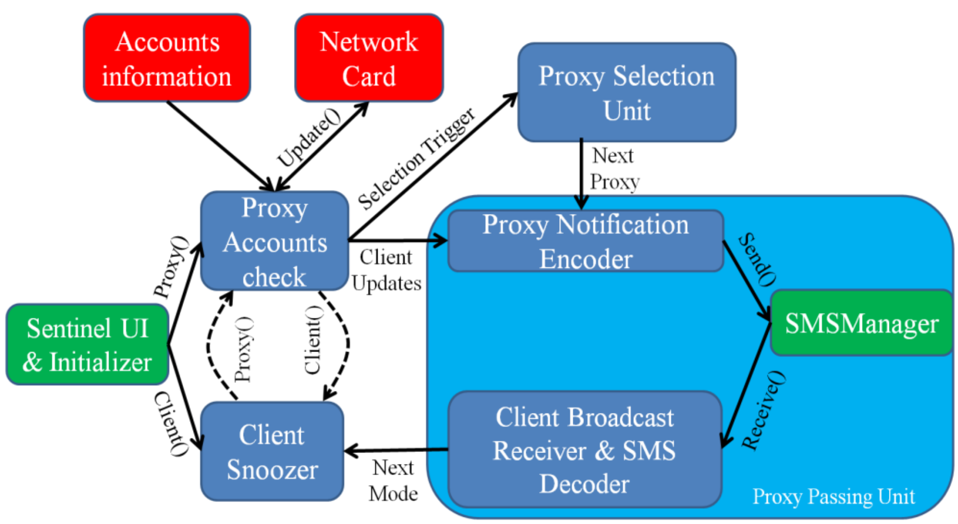

Figure 4. Block diagram of Proxy-Client mode interaction

II. Proxy Selection Mode

The main goal of this research is to investigate various proxy selection methods to produce the most efficient result for group power saving. This proxy selection framework will evaluate several parameters to determine the best phone candidate to operate as a proxy. Currently, several proxy selection techniques have been implemented in this research. These include round robin scheduling, battery level, and notification-frequency schemes. The concept of round robin scheduling is done by rotating proxy responsibilities sequentially through a list that is generated as phones join the group (Figure 5). With this technique, rotation of work among a group of phones can be properly delegated to ensure fairness of workload.

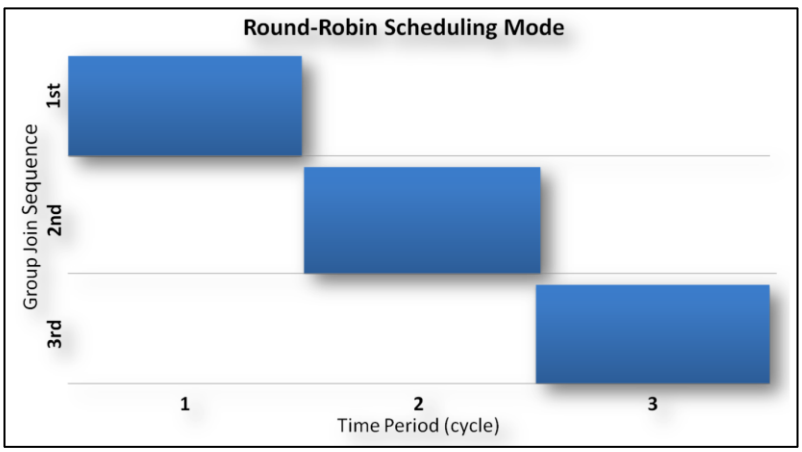

Figure 5. Proxy selection based on Round-Robin Scheduling

The second technique, the notification-frequency scheme determines the next proxy by selecting the user with the most notifications/emails received in a specific time period (Figure 6). By doing so, we can conserve energy by removing the need to turn on hardware components (e.g. screen display, Wi-Fi, and GPS) for the phone with the most frequent notifications. However, in this scheme the role of the proxy would frequently fall on the user with the most notifications/emails.

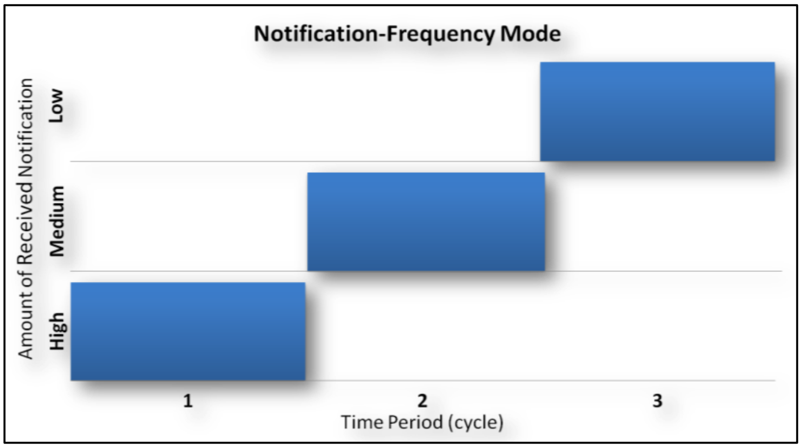

Figure 6. Proxy selection based on Notification-Frequency

Finally, the battery level scheme determines the next proxy based solely on the energy left in each phone. The battery level scheme ensures that the next proxy is not at a critical battery level situation. This allows the chosen proxy device to have sufficient energy to maintain the phone's functionalities. Prioritizing the battery level scheme can help prevent disruption of the system. However, a drawback of this scheme is that the user with the highest battery level has to constantly be the proxy. To further improve this scheme we will introduce an implemented version of the battery percentage mode, the new battery percentage mode will change the length of the proxy period depending on the initial battery level (Figure 7).

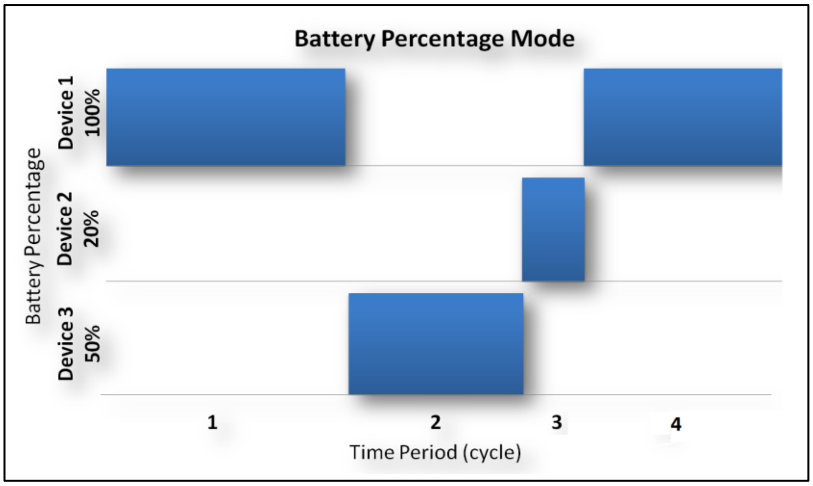

Figure 7. Proxy time slot depending on Battery Percentage

Different situation will require different proxy selection combination. In order to create an algorithm that selects the appropriate proxy selection mode for the situation, we need to study the effects of different proxy selection mode on the smartphone batteries.

III. Experimental Setup

We will first describe the experimental setup and then describe the results in the next section. We implemented the framework on two phones, Galaxy S2 and Galaxy S3. Two smartphones are sufficient to show the general power consumption of the framework in both proxy and client mode. The Galaxy S2 is equipped with Samsung Exynos 4 Dual 45nm with dual core 1.2GHz CPU and 1GB of RAM. The Galaxy S3 is equipped with Samsung Exynos 4 with quad core 1.4GHz CPU and 1.5GB of RAM. Both devices run Android Jelly Bean 4.3 as their operating system. To collect energy data, we use Monsoon external power meter. The power meter has the capability to measure the amount of real time power consumption of the phone, but it is not capable of measuring particular hardware on the phone. To measure the power and energy consumption of each hardware components used by a specific application, we will be using PowerTutor, an android application. We used these devices to conduct two kinds of experiment with the goal of: (i) describing the main differences between the three proxy selection modes and (ii) showing the variation of proxy energy consumption by increasing the number of phones to check in each cycle. In both experiment the estimated power demand for CPU and Wi-Fi was extracted. To measure the battery level, Android has implemented APIs to retrieve battery level. All these information allow us to analyze and have an exact idea of the amount of energy saved.

In the first experiment, we are interested on studying the effects of different proxy selection modes on the smartphone batteries. Each experiment will be 25 minutes long with the initial battery level of GS2 at 100% and GS3 at 40% in all experiments. The proxy selection algorithm runs at periodic intervals which is calculated by divifing the event-time by the number of devices (in this case there is only two devices so the proxy selection will take place after 12.5 minutes). The proxy will check for new notifications with a delay of 1 minute from the end of the last check cycle. We set up a test case where GS2 will be receiving one email every two minutes, while GS3 will only receive one email after 12 minutes of the experiment. There are two reasons to set up the experiment this way: (i) first this allows us to compare notification based mode with the other modes since one user will receives more emails than the other, (ii) secondly we can observe the power consumption causes by the SMS notification. Another assumption is GS2 will always start as a proxy for all the experiment. To set up a baseline for comparison among different proxy selection mode, the round-robin proxy selection mode is implemented on both user devices with no new email for the whole time period of 25 minutes. This gave us an ideal model to observe the proxy energy consumption solely based on the framework itself without any activity and we call this test *no notifications*.

In the last experiment, because testing the proxy-client model in a real group event may require long experimental times, we use real world user traces to setup two different group events. We use 34 users traces of smartphones recorded for an entire year. These traces were created from a previous research study [4] and are available online in SQL database file. One of these files report how users interact with the phones during each day. The data available are 1) what application the user is opening, 2) the timestamp, and 3) the length of the interaction. In later section we describe in more details how we use these information to emulate group event.

Chapter 3: Results and Discussion

In this section, we evaluate the performance of proxy-client mode considering the email synchronization. First we examine the result of power consumption with three proxy selection mode. Then we evaluate the potential energy savings of client mode. Finally, we use real world traces data to setup two case studies and compare the energy usage of smartphones with baselines.

I. Proxy Selection Mode

The experimental results and evaluate the power consumption of three proxy selection modes (round robin, battery level, and notification frequency) as mention in the previous section is explain in this section. Figure 8 and 9 show the results of CPU, Wi-Fi, and total power consumption of GS2 and GS3 based on different proxy selection mode. Figure 10 shows the consequent battery consumption.

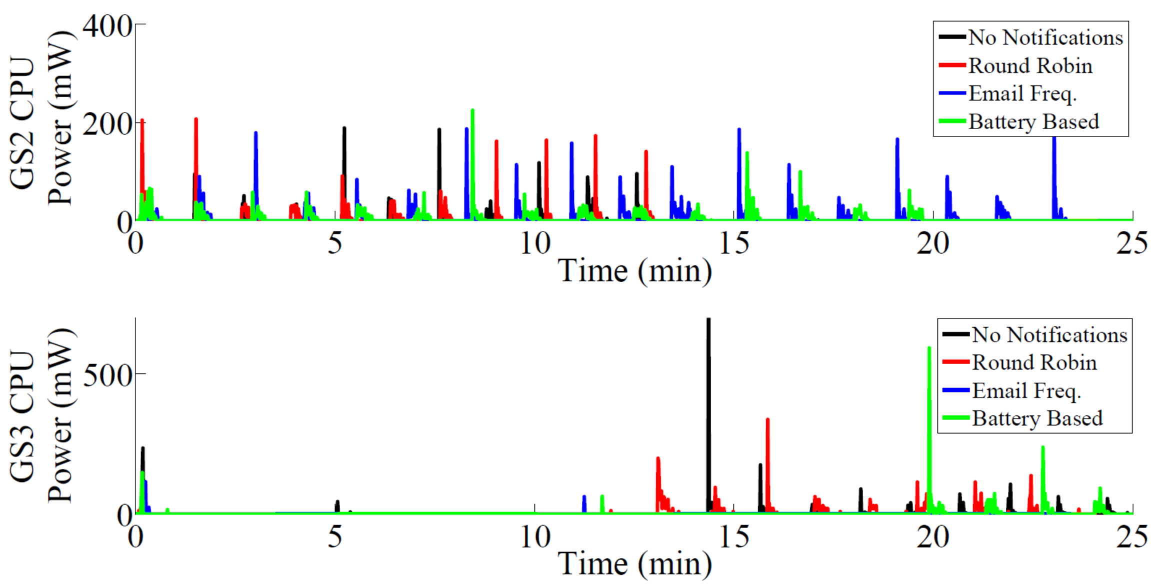

Figure 8. CPU power demand with different proxy selection mode

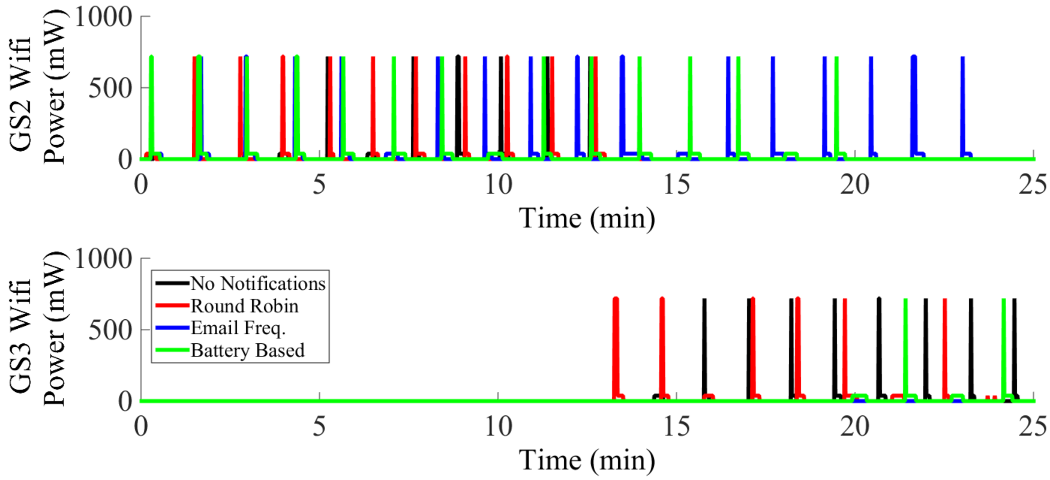

Figure 9. WiFi power demand with different proxy selection mode.

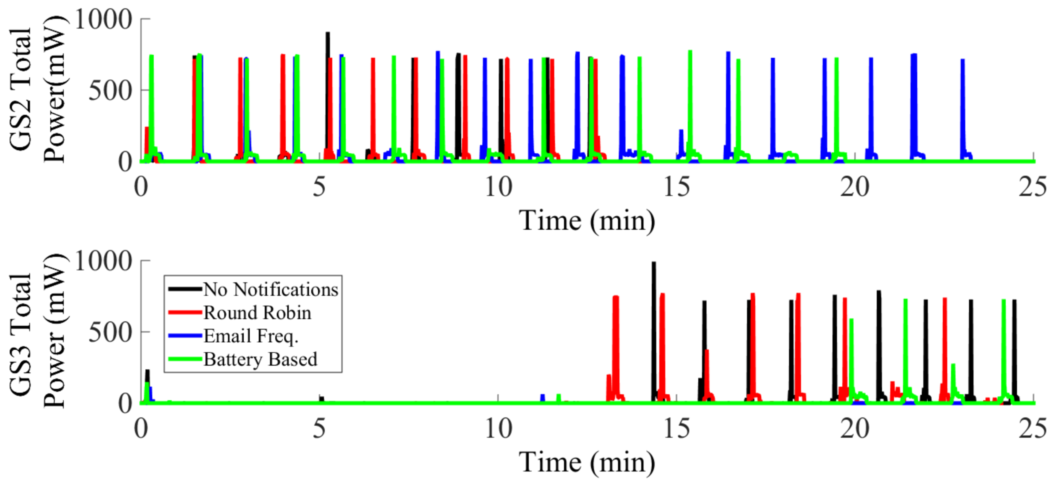

Figure 10. Total power demand with different proxy selection mode.

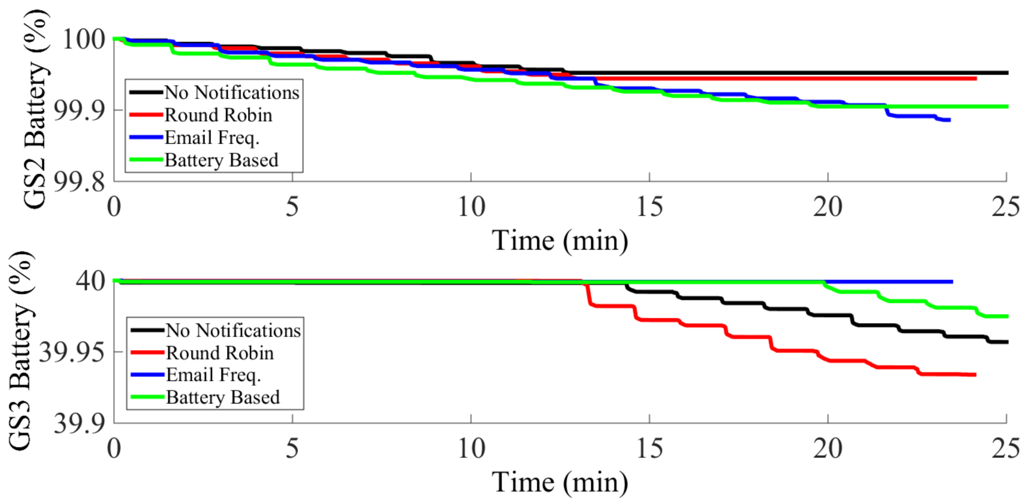

Figure 11. Total battery consumption with different proxy selection mode.

As shown in the figure, the round-robin mode selects GS2 as the proxy for the first 12.5 minutes and then switches the proxy to GS3 for the next 12.5 minutes. As shown in Figure 11 (comparing *no notification* and round-robin), sending or receiving new notification does not cause any huge difference in energy consumption. Our hypothesis is that this is because while running under notification frequency mode, the switching of proxy does not necessarily take place. Since the GS2 user receives more emails during the first period than the GS3 user, the GS2 remains as the proxy throughout the whole event. Even though notification frequency mode shows the least power consumption, it is unfair to the proxy user since the proxy will be the same throughout the whole event. Lastly, we evaluate the battery percentage mode since GS2 has 60% more battery level than GS3. Therefore GS2 will run as the proxy for 20 minutes instead of only 12.5 minutes, while GS3 run as the proxy for 5 minutes until the group event ends. As a result, at the end of the group event the GS3 shows higher battery level in notification mode (during which GS3 was never proxy) as compared to round robin mode. On the other hand, the GS2 consumes more power in notification mode as compared to round robin mode since the proxy period is longer, but does not consume as much power as compared to notification frequency mode since it can take advantage of the snooze state. This ensures a promising fairness system due to two facts: (i) all the smartphone are assign to be the proxy at least one period, and (ii) the length of the period is proportional to the initial battery level. These experiments have showed that the proxy selection for battery percentage algorithm shows best trade-off between battery consumption and fairness among the three proposed algorithm.

Most importantly, this experiment shows that the presence of new notifications does not affect the total power consumption of the proxy by very much. The proxy mode shows only a little overhead on the battery consumption since it deplete at most the 0.1% of battery after 15 minutes of notification checking.

II. Client Energy Savings

In this section we will discuss how energy saving can be achieve by using this framework. For this experiment the Monsoon power meter is connected to the power pins of the Samsung Galaxy S3. We will first compare the difference of power consumption between proxy and client mode. For this experiment only one smartphone is sufficient to show the energy savings because each client only communicates with the proxy and each one saves the same amount of energy after joining the framework. We start recording the power consumption with the phone screen off and both Wi-Fi and 4G on. Figure 12 shows the power demand of GS3. The average power consumption of the smartphone is about 379mW over a period of 15 minutes. After 15 minutes, there is a spike in power consumption due to the necessity to turn-on the screen and switch it to client mode (low power by turning off all data connection except for basic GSM). The power consumption after switching to client mode was reduced to about 122mV over 15 minutes period, which corresponds to about 68% in power reduction. Figure 13 shows a comparison of energy consumption of GS3 with and without data capabilities turned on. The two curves are quite linear and the difference in the slope leads to about 70% of energy reduction. These results prove that a client phone can save a lot of energy using this framework during group events.

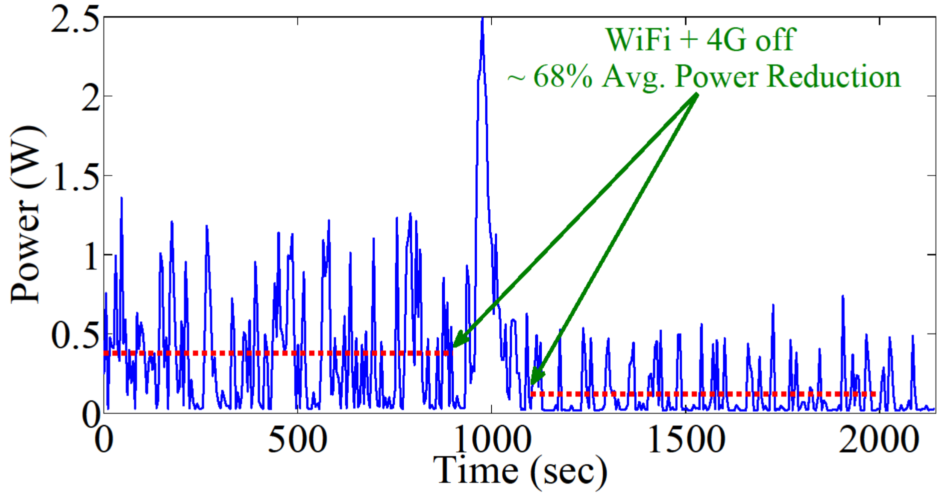

Figure 12. Switching from proxy to client mode shows a significant power reduction.

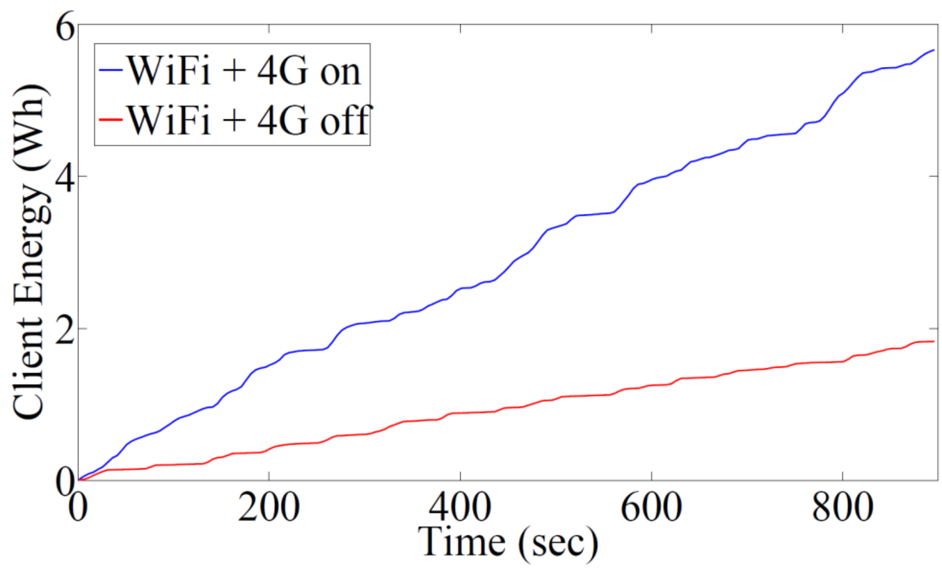

Figure 13. Energy comparison of data connections on and off.

III. Emulation of Group Event

In this section, we describe 3 different scenarios and compare the results:

- Baseline scenario: Normal usage of smartphone with Wi-Fi and 4G data connections connected at all times.
- Individual scenario: The user turn on data connections to synchronize and check for email notifications every 15 minutes, but energy may still be wasted if there is no email to update during the synchronization.
- Proxy-client mode: Phones operate under the proposed consolidation system.

To analyze and compare the total energy saving with these 3 scenarios, we will be measuring their performance based on an email application. First, we randomly choose one data trace to compare the energy savings between the proxy-client model and the baseline scenario. Then we choose 5 random data traces and we use them to emulate two group events at meeting time (10:00 – 12:00) and party time (18:00-24:00).

First, we use MYSQL to select from the database a random user's email interactions for an entire year [4]. In particular we extract the interactions that are shorter than 10 seconds, which as we mentioned in the previous sections, we classify as email events that lead to a waste of energy [5]. To clarify, our proxy-client model is built with the intention to minimize this waste. For both baseline and individual scenario, we assume that email automatically synchronizes every 15 minutes. However, the baseline scenario will have data capabilities turned on at all times whereas the individual scenario will manually turn on data capabilities every 15 minutes to synchronize the email and then turn off data afterwards. To calculate the energy consumed, we use the power consumption of Wi-Fi and network data measured with the Monsoon power meter. This information is show in Figure 12. This lets us compare the energy consumed in the baseline scenario with the energy saved by the individual scenario and proxy-client model which turns off data connections between synchronization. Then we take the number of wasteful email interactions to further calculate the energy savings of the proxy-client model which would avoid the wasteful synchronization. Figure 14 shows the results in terms of energy usage over the 24 hour day. As shown in the graph, the energy consumption of the proxy-client method is 2J less than the baseline scenario and on average 200mJ less than the individual scenario. The result of the individual scenario compared to the proxy-client method, however, is heavily dependent on the users interaction with the email app. The more interactions, the higher the probability of useless interactions which will leads to more savings.

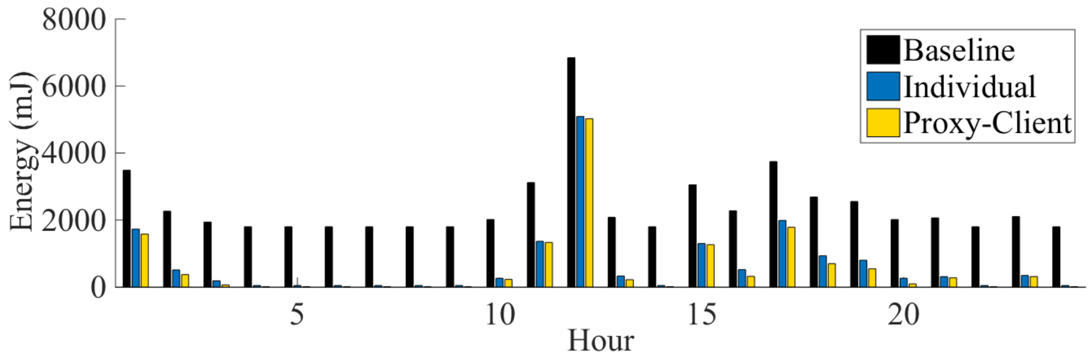

Figure 14. Average of email application usage for one year with different user

To evaluate the proxy-client model in a realistic situation, we use the data traces of five random users for one random day of the year. Two different events are created, a meeting between 10:00 to 12:00 and a party between 18:00 to 24:00. During these two time periods we extract the email interactions of less than 10 seconds and we calculate the energy savings of the proxy-client method as described in the previous paragraph. We randomly select one of the users to be the proxy (user 5 in meeting and user 3 in party) and we add to the selected user the energy overhead of the proxy calculated in previous section. Figure 15 and 16 shows the results. During the meeting group event, the proxy-client method allows the reduction of on average about 56% of the energy consumption with respect to the baseline scenario and about 7% with respect to the individual scenario. During the party event, the proxy-client method on average is able to reduce about 75% of the energy consumption with respect to the baseline scenario and about 33% with respect to the individual scenario.

These results show that the proxy-client method leads to a significant reduction of energy consumption for a group of smartphones, by offloading activities on one single phone.

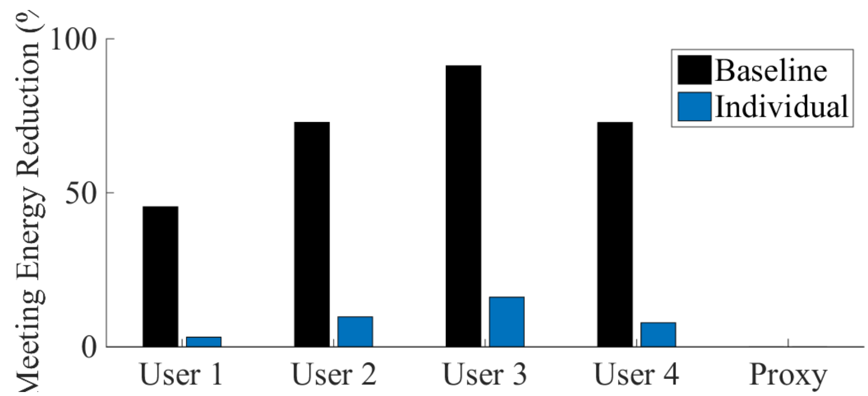

Figure 15. Potential energy savings of five random users during a meeting group event.

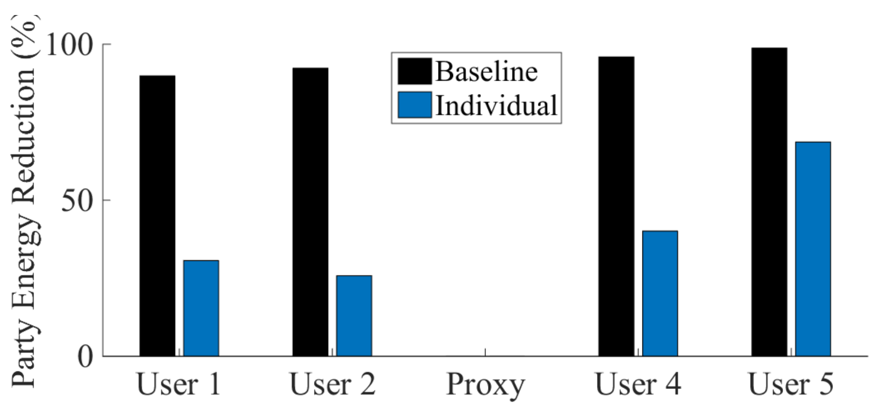

Figure 16. Potential energy savings of five random users during a party group event.

Chapter 4: Conclusion and Future Work

To study the problem of group proxy selection, we propose a consolidation framework within a group of smartphones to reduce overall energy consumption. The consolidation framework is prototyped in two Android devices to test for real user traces and to verify its energy savings. By utilizing a combination of different proxy selection modes, energy savings of up to 75% has been achieved from experimental results, compared to normal usage of individual smartphones.

For future experiment, we propose to analyze the minimum overhead requirement of the proxy to ensure no Quality of Service degradation. In addition, a self-sustainable system is needed to monitor the user behavior and performs maintenance operation dynamically. Lastly, as for the privacy concerns, we hope to work on implementing proper security encryption system.

.

References

1. Y. Chon, E. Talipov, H. Shin, and H. Cha, “Mobility prediction-based smartphone energy optimization for everyday location monitoring,” *SenSys*, 2001.
2. B. Chun, S. Ihm, and P. Maniatis, “Clonecloud: Elastic execution between mobile device and cloud,” *EuroSys*, 2011.
3. T.Das, P. Padala, and V. N. Padmanabhan, “Litegreen: Saving energy in networked desktops using virtualization,” *Usenix*, 2010.
4. C. Shepard, A. Rahmati, C. Tossell, L. Zhong, and P. Kortum, “Live-lab: measuring wireless networks and smartphone users in the field,”*SIGMETRICS*, 2010.
5. T. Wasserman, “Email takes up 28http://mashable.com/2012/08/01/email-workers-time/.
6. R. Balani, “Energy Consumption Analysis for Bluetooth, WiFi, and Cellular Network”. Tech. no. TR-UCLA-NESL-200712-01. Networked & Embedded System Laboratory, 2007.